

Oficinas Ingrediente X

Apostila do aluno

Autores:

Aida Araújo Ferreira, Welton Pereira da Luz Felix, Fábio de Lima Ferreira Papais, Ryan Gomes Paiva, Estevão Pereira da Silva, Luciano Alves Filho, Irwing Joshua Nery da Silva, Cristiane Lucia da Silva, Gilmar Gonçalves de Brito, Ioná Maria Beltrão Rameh Barbosa e Vânia Soares de Carvalho

**Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)**

Oficinas ingrediente x [livro eletrônico] :
apostila do aluno. -- 1. ed. -- Recife, PE :
Ed. dos Autores, 2022.
PDF.

Vários autores.
ISBN 978-65-00-59785-1

1. Informática (Ensino fundamental)
2. Linguagem de programação (Computadores)
3. Programação (Computadores) - Estudo e ensino
4. Scratch (Linguagem de programação para computador).

23-140804

CDD-372.358

Índices para catálogo sistemático:

1. Informática : Ensino fundamental 372.358

Aline Grazielle Benitez - Bibliotecária - CRB-1/3129

O que são as oficinas Ingrediente X?



As oficinas Ingrediente X têm o objetivo de introduzir e ensinar **programação para estudantes do 8º e 9º ano**. Para isso, vamos utilizar a linguagem de programação e plataforma chamada **Scratch**, criada pelo MIT para iniciar em programação.



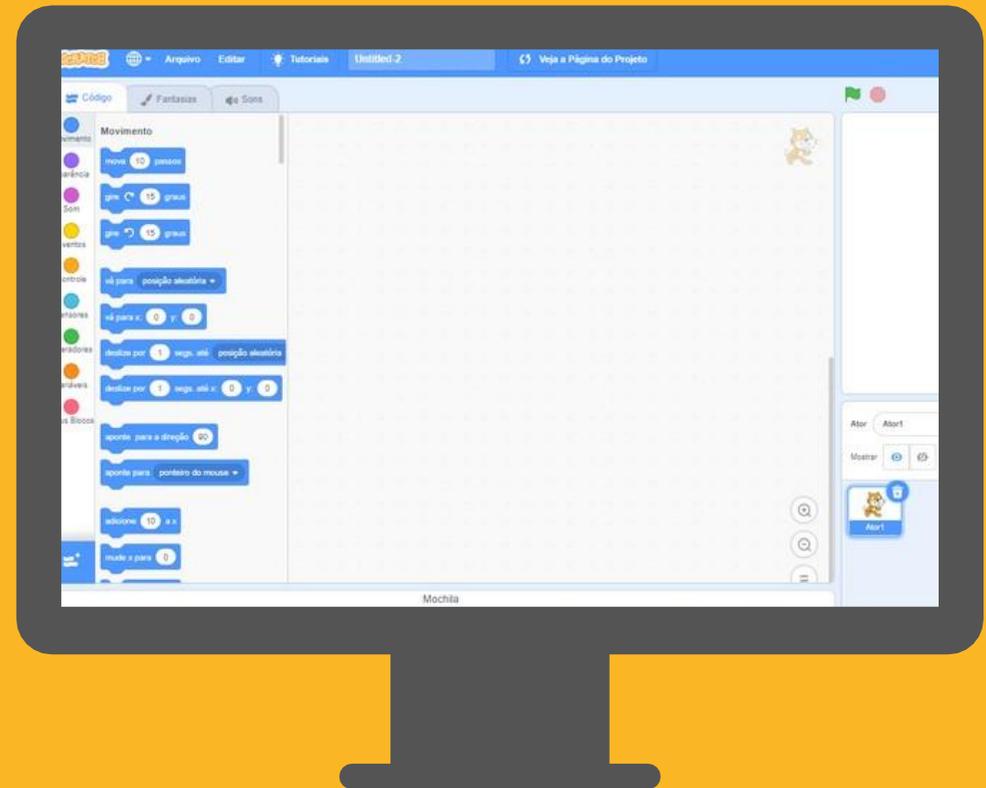
Durante as oficinas, vamos incentivar o **trabalho ativo e em grupo**, além da criação de projetos utilizando a ferramenta, como jogos, animações e histórias que explorem outras áreas do conhecimento, como português, arte, literatura, matemática, ciências, geografia e história de forma **interdisciplinar**.

Módulo 1

Conhecendo o editor do Scratch

Conteúdo

1. O que é o Scratch?
 - a. O que fazer com o Scratch
2. Interface do Scratch
 - a. Entendendo blocos
3. Seu primeiro programa
 - a. Blocos iniciais
 - b. Entrada e saída de dados



O que é o Scratch?

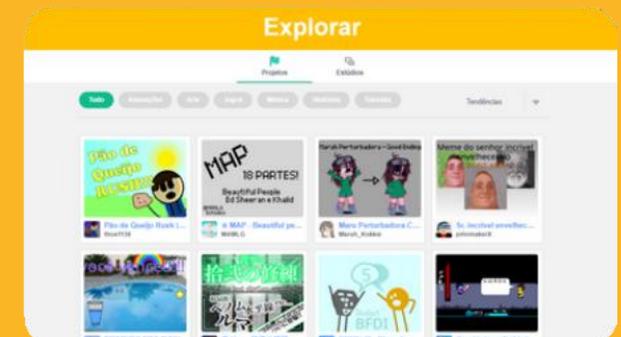


E o que pode ser feito com ele

O Scratch é uma linguagem de programação criada com o intuito de **ensinar lógica de programação** para adolescentes de 8 a 16 anos de idade.



Hoje em dia, além da linguagem de programação e do site onde podemos fazer nossos programas, o **Scratch se tornou uma comunidade** na internet com diversas pessoas e projetos.



Pare e pense:

Mas o que é uma linguagem de programação?



Bom, então vamos começar apenas por “**Linguagem**”.

O que é uma linguagem? De forma simples, é como nós comunicamos nossas ideias e sentimentos às pessoas. Por exemplo, a fala e a escrita são dois tipos de linguagem.



Já quando pensamos em **linguagem de programação**, o que nos vem a mente são computadores, e não pessoas, certo?



Então linguagens de programação são como nós dizemos aos computadores o que nós queremos que eles façam. São **conjuntos de instruções sobre o que o computador deve fazer**, como se fosse uma receita de bolo, por exemplo.

Ao lado nós temos um exemplo de uma linguagem de programação chamada **Java**. Esse código está mandando o computador escrever **"Olá Oficinas Ingrediente X!"** na tela.

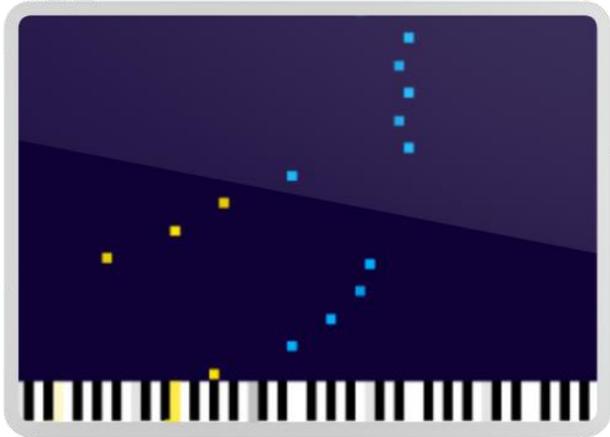
```
class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Olá Oficinas Ingrediente X!");  
    }  
}
```

Mas.... isso ainda é muito complicado e difícil de entender, não é mesmo? É pra isso que existe o Scratch, veja como fazer o mesmo usando o Scratch:



Viu? Agora está um pouquinho mais fácil de entender, e no mínimo mais amigável, não é mesmo? A gente vai entender exatamente como esse bloco de código do Scratch funciona mais pra frente.

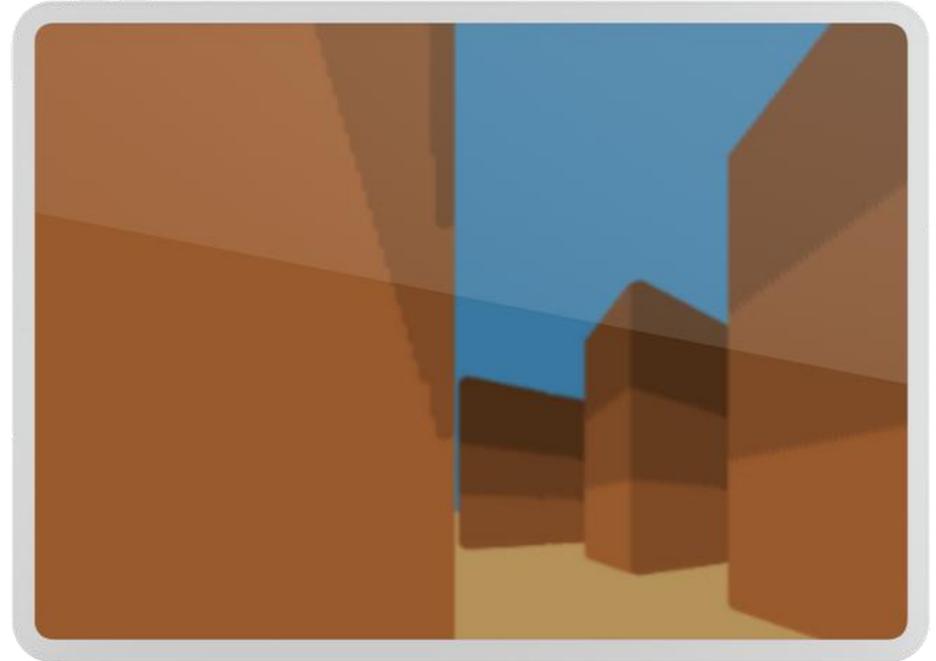
Veja o que pode ser feito com o Scratch



Músicas



Histórias animadas



Jogos

E muito mais! Você pode inclusive fazer o programa interagir com sua câmera ou microfone, por exemplo.

Interface do Scratch

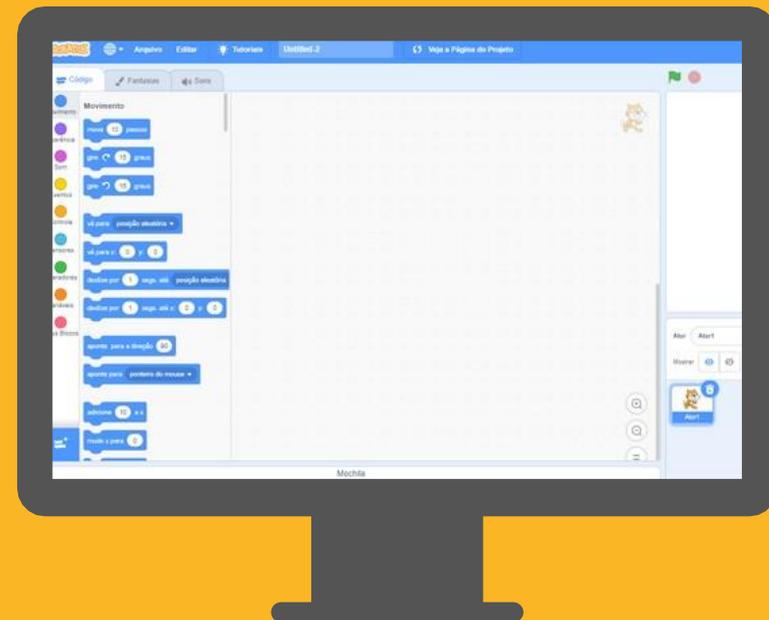


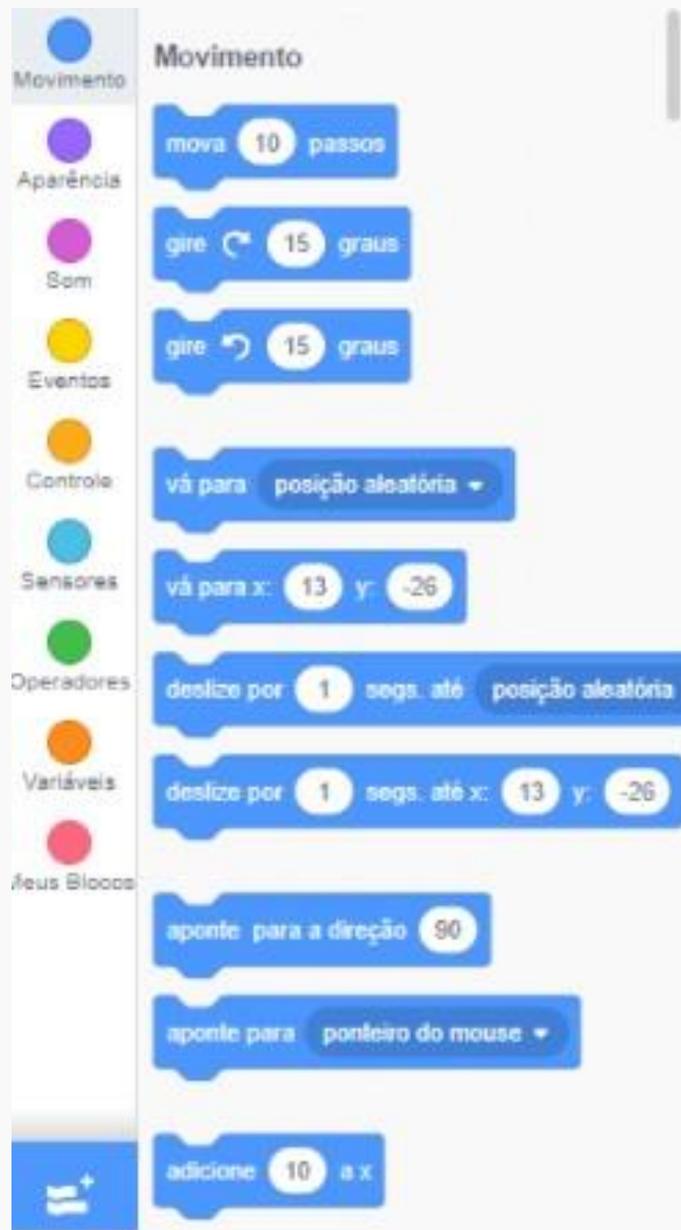
Entendendo os blocos

Para utilizarmos o Scratch, vamos acessar o site <https://scratch.mit.edu/>, e pode ficar tranquilo pois ele é inteiramente gratuito!

Se tiver acesso à internet e um computador ou celular, **tente acessar** e acompanhar esta seção com o site aberto.

Por enquanto estamos apenas na página principal, mas para acessarmos o editor de código devemos clicar em "**Comece a criar**". Se você quiser, depois pode explorar as outras abas e partes do site.





Do lado esquerdo temos a área em que podemos ver os blocos que vamos usar para fazer os programas. Eles estão separados em categorias (Movimento, Aparência, Som, Eventos, Controle, etc). Você também pode rolar o mouse e ver todos os blocos disponíveis para usar.

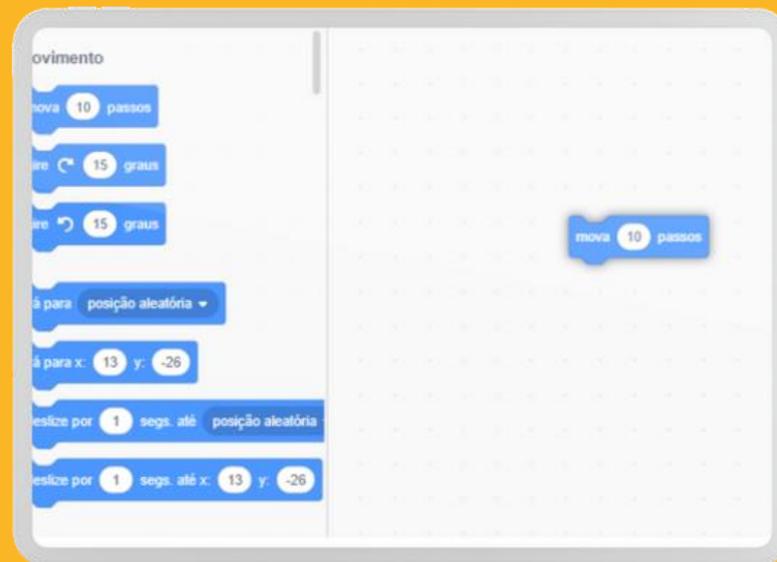
Veja que estamos na aba **"Código"**, que é onde vamos passar a maior parte do tempo. Nas outras duas abas "Fantasias" e "Sons" podemos fazer algumas coisas que vamos ver no final do curso.

⋮

Ainda são muitas partes diferentes, mas pode ficar tranquilo que vamos aprender tudo ao longo das Oficinas!



Logo ao lado, temos este espaço em branco, que será a nossa área de trabalho! Tente pegar e arrastar qualquer um dos blocos para esta área. Veja que a agora você tem um novo bloco, e pode colocar quantos blocos quiser.



Se tentar pegar outro bloco e colocar perto de um bloco que já estava lá, você pode perceber que aparece uma sombra. **Se soltar o bloco, ele fica grudado com o outro!** É assim que funciona a interface do Scratch, com **blocos que se encaixam em outros, como se fossem peças de montar.**

Tente encaixar novos blocos com outros para **se sentir mais á vontade com a plataforma.**

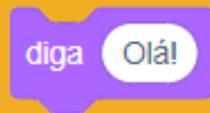


Hello World

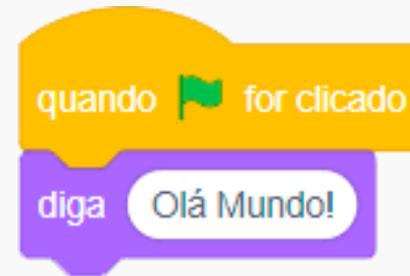


Fazendo seu primeiro programa no Scratch

Uma das primeiras coisas que se aprende em qualquer linguagem de programação é a criar um programa bem simples que imprima na tela a mensagem "**Olá Mundo**". É exatamente o que faremos a seguir. Você precisará destes blocos para criar o programa:



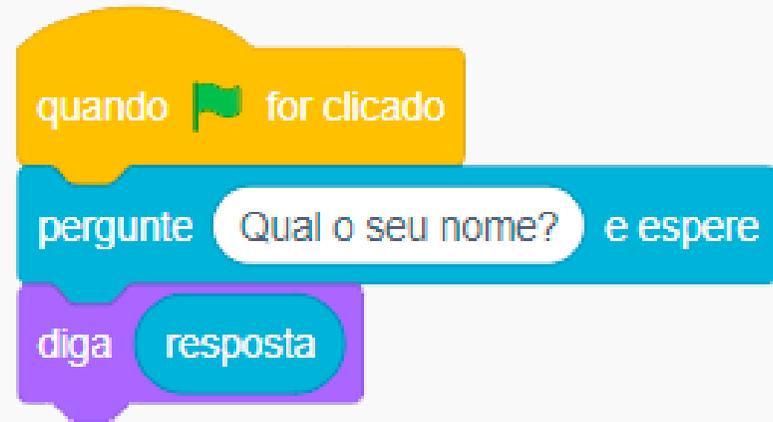
**Encaixe-os e mude o texto.
Escreva "Olá Mundo!"**



Ao arrastar os blocos e criar o programa acima, **toda vez que a bandeira for clicada o gato irá dizer "Olá Mundo!"**. É possível alterar o texto, e toda vez que a bandeira for clicada, o gato vai falar o que está escrito.

Esse programa ainda pode ser aprimorado, e é o que faremos a seguir...

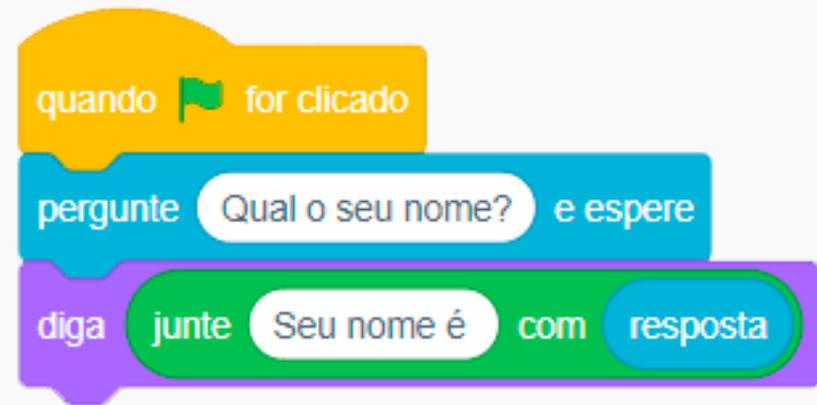
O programa anterior imprime mensagens que o programador coloca. Mas e se quisermos perguntar algo ao usuário e **mostrar o que ele escreveu**? É isso que faremos a seguir no programa ao lado.



Com o programa acima, toda vez clicamos na bandeira para rodá-lo, será perguntado ao usuário **qual o nome dele** e em seguida ele imprime o nome que o usuário digitou. Se antes tínhamos um programa que apenas imprimia uma mensagem colocada pelo programador, agora ele **interage com o usuário e imprime algo que ele escreveu**. Ainda é possível modificar este programa e combinar o que vimos agora com o que vimos anteriormente: o programa imprime algo que o usuário digitou + algo que o programador digitou. Faremos isso a seguir...



Podemos aprimorar o programa anterior fazendo com que ele pergunte algo ao usuário e na hora de imprimir junte esse dado com algo que o programador digitou. É isso que faremos.



Ao rodar o programa acima, ele vai perguntar qual o nome do usuário e vai juntar a frase "Seu nome é" com o nome que o usuário digitou. Esse ato de juntar textos se chama **concatenação**.

Nesse módulo, aprendemos a imprimir textos no Scratch; fizemos nosso primeiro programa que imprime "Olá Mundo" e modificamos ele duas vezes para aceitar e imprimir dados que o usuário digita.

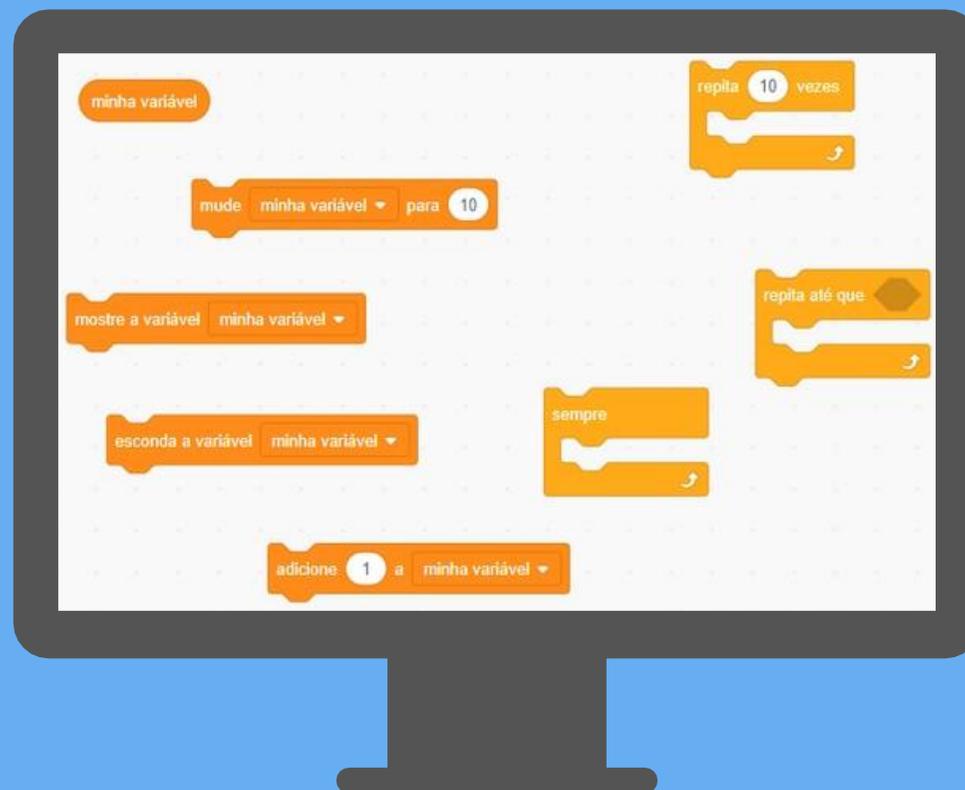


Módulo 2

Variáveis e estruturas de repetição

Conteúdo

1. Variáveis
2. Loops
 - a. Repita... vezes
 - b. Repita até que...
 - c. Sempre



Variáveis



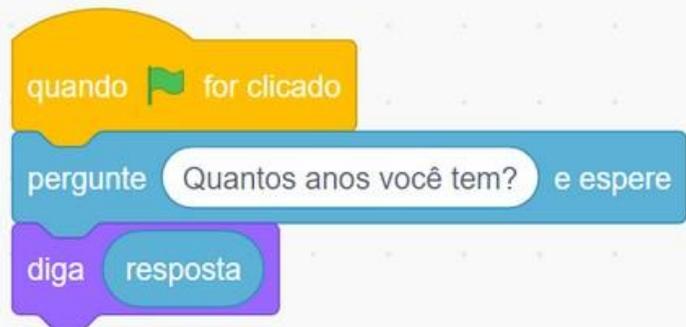
Guardando valores do programa

No módulo anterior, fizemos o gatinho perguntar o nome do usuário e depois repeti-lo. O nome digitado pelo usuário foi guardado em um bloco chamado **resposta**.

Esse bloco é o que chamamos de variáveis.

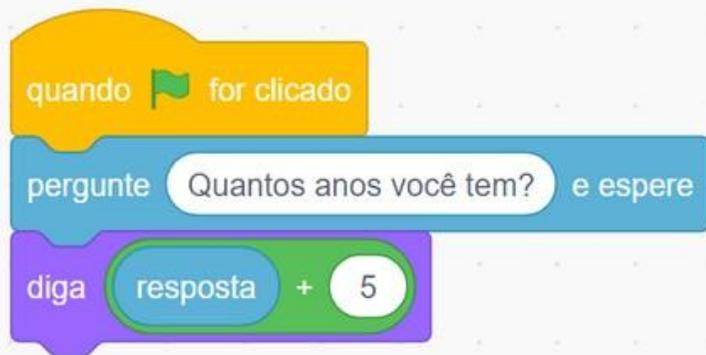
As variáveis são valores do nosso programa que guardamos na memória do computador. Elas podem ser números, textos ou ainda um valor lógico (verdadeiro ou falso). Com elas, podemos fazer cálculos, receber e manipular dados informados pelo usuário ou ainda controlar estruturas do programa.

Vamos começar com algo parecido com o exemplo do módulo anterior, perguntando a idade do usuário.



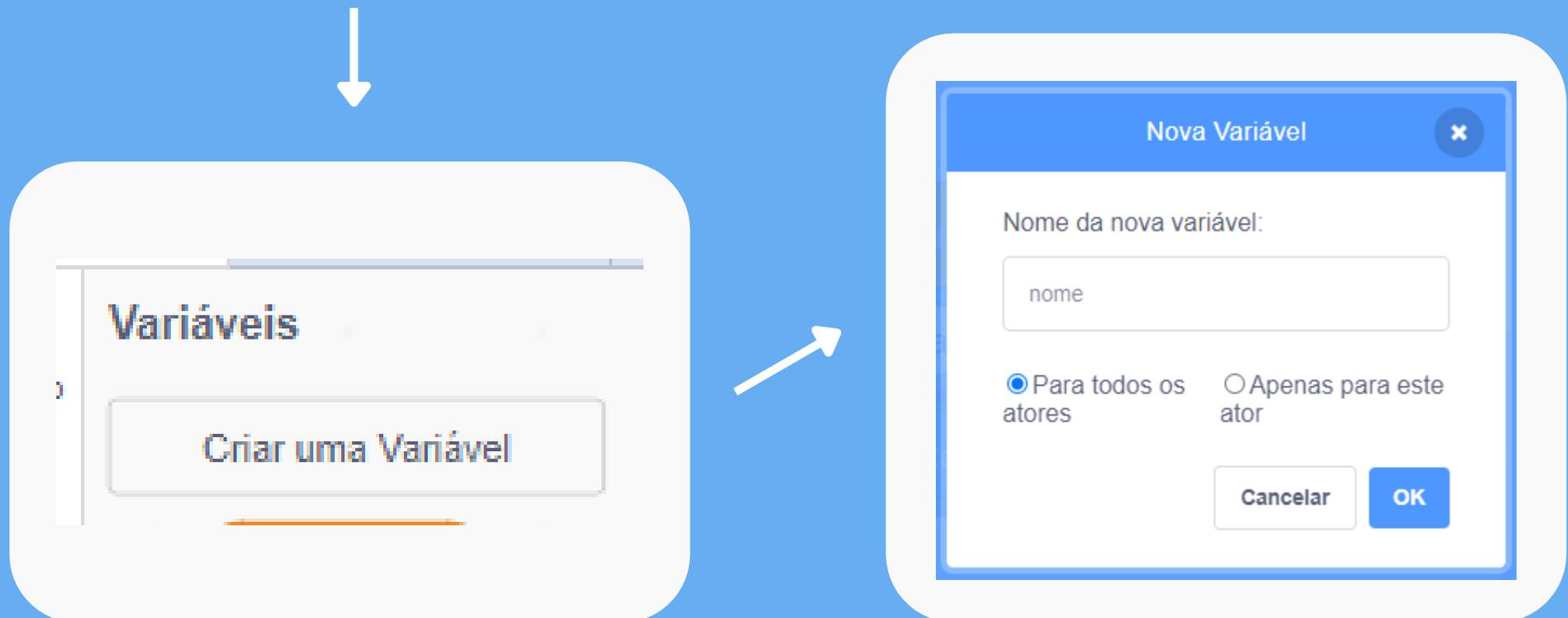
Vamos então usar a variável de resposta, com a idade informada para indicar quantos anos essa pessoa terá daqui a 5 anos.

Para isso, vamos colocar no bloco **diga** uma operação de soma, somando 5 à idade atual.



Agora que já vimos como utilizar variáveis, que tal perguntarmos o nome e a idade da pessoa e depois fazermos o gatinho escrever uma saudação?

Para fazermos mais de uma pergunta e guardarmos cada uma das respostas, criamos uma variável personalizada em “Criar uma variável”. Depois basta inserirmos o nome dela e clicarmos em “ok”.



Vamos criar duas variáveis: **nome** e **idade**.

Agora vamos atribuir a resposta de cada pergunta à sua respectiva variável.

Podemos usar o bloco **junte... com...** para criarmos o texto da frase final.



```
quando for clicado
pergunte Qual o seu nome? e espere
mude nome para resposta
pergunte Quantos anos você tem? e espere
mude idade para resposta + 5
diga junte junte Olá, com junte nome com ! Daqui a 5 anos você terá com junte idade com anos.
```

The image shows a Scratch script on a grid background. It starts with a yellow 'quando for clicado' block. This is followed by a blue 'pergunte' block with the text 'Qual o seu nome?' and 'e espere'. Below it is an orange 'mude' block with 'nome' in a dropdown menu, 'para' in the middle, and 'resposta' in a blue input field. The next is another blue 'pergunte' block with 'Quantos anos você tem?' and 'e espere'. This is followed by an orange 'mude' block with 'idade' in a dropdown menu, 'para' in the middle, 'resposta' in a blue input field, a plus sign, and the number '5'. The final block is a purple 'diga' block containing a green 'junte' block with 'junte' in a dropdown menu, 'Olá,' in a white input field, 'com', another 'junte' block with 'nome' in a dropdown menu, 'com', a white input field containing '! Daqui a 5 anos você terá', 'com', a third 'junte' block with 'idade' in a dropdown menu, 'com', and a final white input field containing 'anos.'



Loops



Repetindo um trecho de código automaticamente

É bem comum querermos que uma parte do nosso código seja executada mais de uma vez seguida.

Vamos pensar no exemplo a seguir:

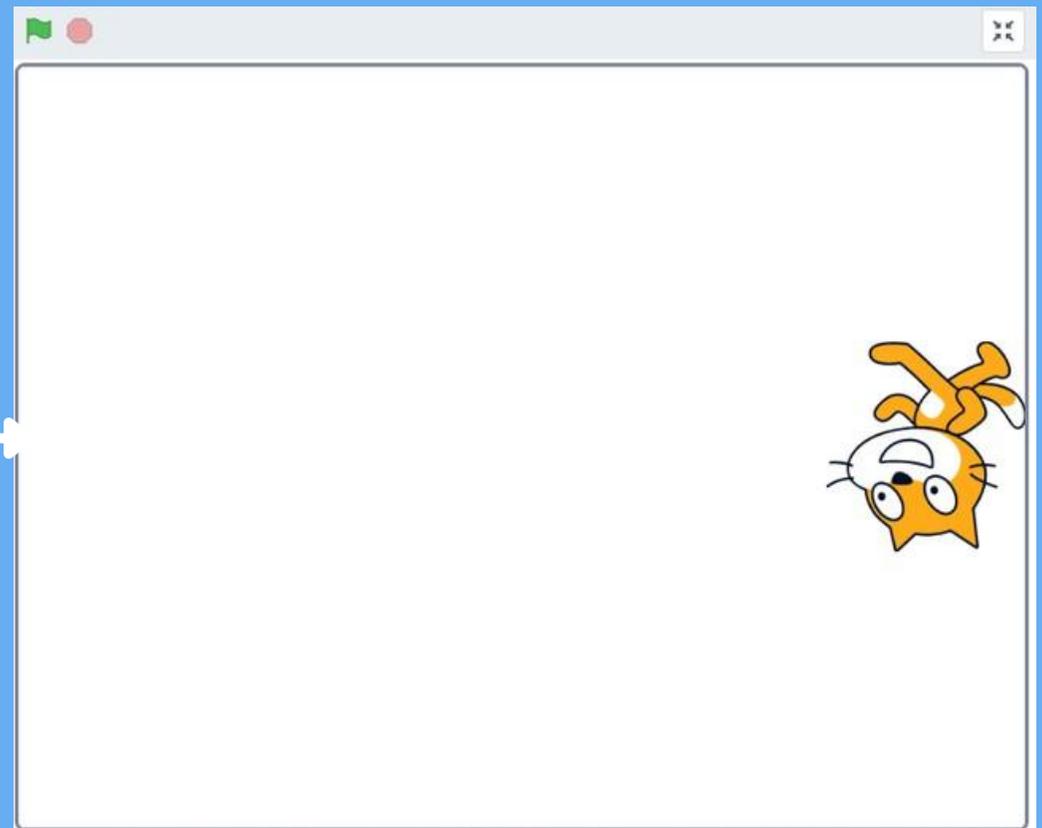
Vamos fazer com que o gatinho se movimente um pouco. Para começar, vamos fazê-lo dar 50 passos. Podemos utilizar o bloco  e colocarmos o número 50. Isso já deve funcionar!



Porém se clicarmos na bandeira verde várias vezes, podemos ver que o gatinho vai para fora da tela. Para resolver isso, vamos adicionar o bloco

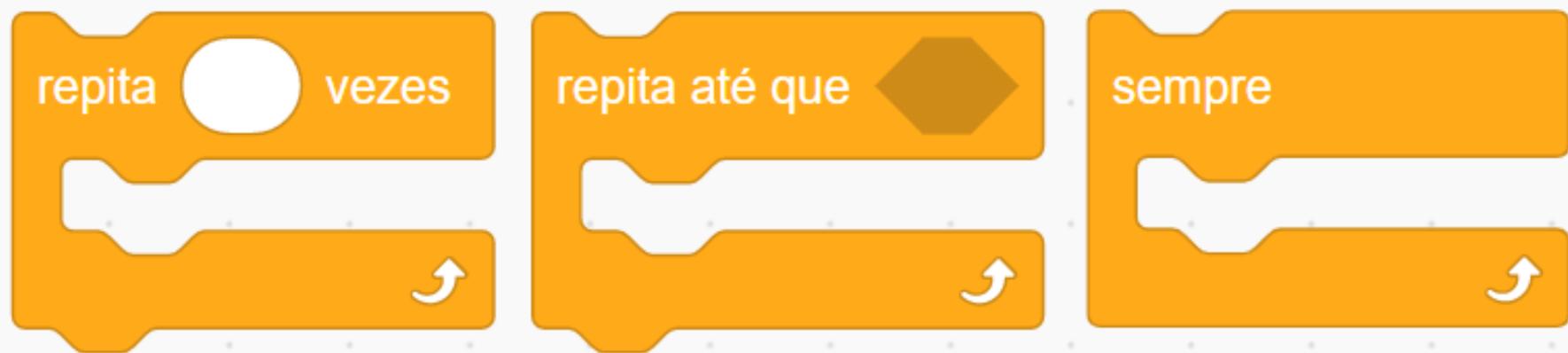
se tocar na borda, volte

. Agora esse problema foi resolvido.

A white rounded rectangle containing three Scratch code blocks stacked vertically. The top block is yellow and says "quando  for clicado". The middle block is blue and says "mova  passos". The bottom block is blue and says "se tocar na borda, volte".

Mas e se quisermos fazer com que ele ande mais vezes sem precisarmos clicar na bandeira verde toda vez? É aí que podemos usar as estruturas de repetições.

As estruturas de repetição, também chamadas de laços de repetição, laços de iteração ou *loops* são estruturas presentes nas linguagens de programação que nos permitem executar um trecho de código múltiplas vezes.



Loops

Repita... vezes

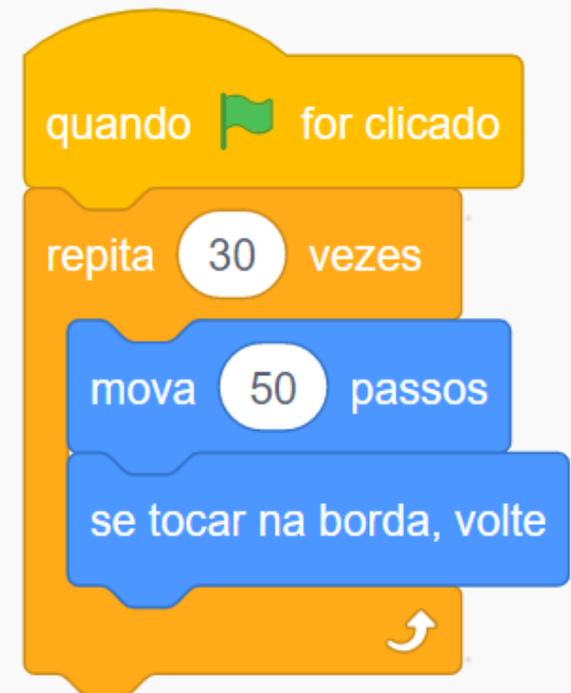


Voltando ao nosso exemplo, se queremos que o gatinho repita o processo de andar 50 passos e voltar caso encoste na borda 30 vezes, podemos envolver

esse trecho do código com o bloco



e indicar a quantidade de repetições.



Loops

Repita até que...



Seguindo a mesma lógica do caso anterior, vamos fazer com que o gatinho se mova até que o usuário escreva "pare".

Para isso, vamos juntar os conceitos de variáveis que vimos anteriormente com a estrutura



Loops

Sempre



Por último, se quisermos que o gatinho fique andando de um lado para o outro sem parar,

podemos utilizar a estrutura



Ela vai fazer com que o código se repita infinitamente.

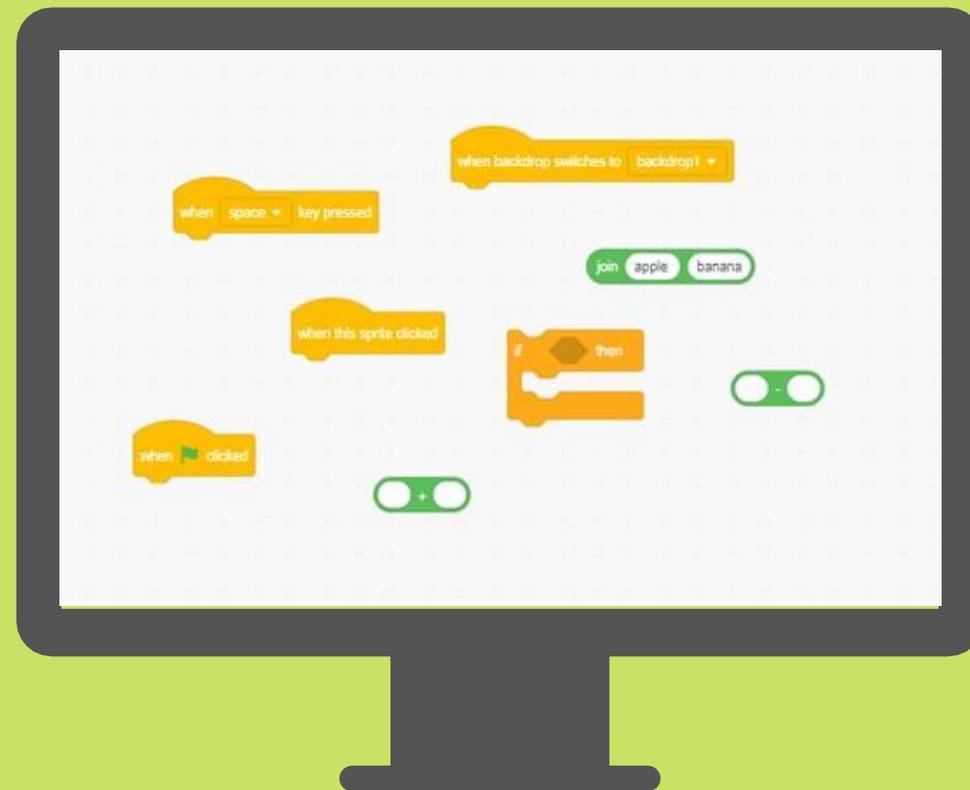


Módulo 3

Condicionantes e operadores

Conteúdo

1. O que são condicionais?
2. Blocos If
3. Blocos if-else



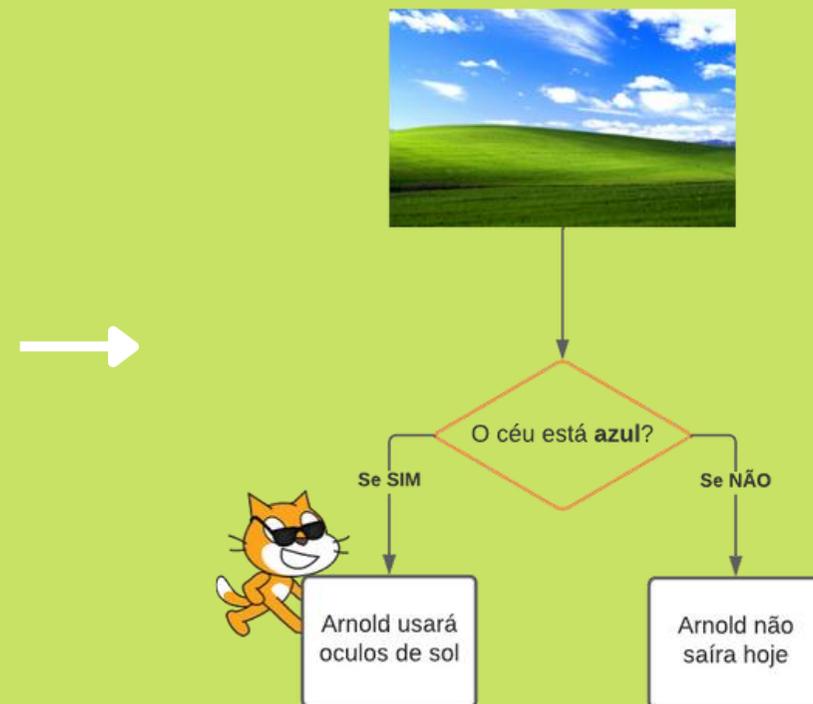
O que são condicionais?



Adicionando condições

Estruturas condicionais são blocos que permitem dentro do seu jogo/código que você verifique logicamente (Booleano) casos verdadeiros e falsos. A partir destas condições você pode organizar seus blocos para executar estas verificações para que outras partes do código funcionem. Dentro do seu scratch, você encontrará os blocos de condicionais dentro da opção Controle no menu da lateral esquerda. Abaixo um diagrama explica o funcionamento das condicionais no geral.

Dá uma olhada como é o raciocínio das condicionais



Bloco If ou Se-Então



Iremos aprender o seguinte bloco.
Você pode encontrá-lo na aba
Controle



O if-then pode ser traduzido com “se (argumento) for verdadeiro, então faça o seguinte:”. Podemos chamar isso de uma condicional simples, onde passamos um argumento, o que significa que iremos usar um bloco Scratch como a parte que será checada para prosseguir o código, então várias condições de blocos Scratch podem ser analisadas em blocos If, como toque, tecla pressionada, posição do mouse. Então, para entendermos como usaremos isso, vamos observar um exemplo:



Neste exemplo temos 4 tipos diferentes de blocos.



Inicialização

Sensores

Movimento

Condicional

Basicamente, a lógica por exemplo de realizar um pulo no Scratch ou mover o personagem para direita ao pressionar uma tecla consiste através destes 4 blocos em:

Assim, fica fácil identificar que estes blocos condicionais podem ser usados para gerar interações caso os argumentos que passamos se verifiquem usando argumentos lógicos, cronômetros, teclas e etc.

Inicializar



Inserir condicional



Checar argumento



Realizar ação



Blocos If-else ou se-então-senão



Iremos aprender o seguinte bloco. Você pode encontrá-lo na aba Controle



Como em condicionais buscamos situações que um argumento seja verdadeiro para que executemos nosso código, o que fazer quando não podemos antecipar que um argumento será verdadeiro? É baseado em situações problema como esta que o bloco **Else** pode ser utilizado. Seu funcionamento é basicamente adicionar uma opção ou uma exceção caso o argumento do bloco **If** não seja satisfeito, então vamos pensar em como transformar nosso bloco **if** em um **if-else**.



No exemplo acima, vemos uma tentativa de uso do bloco if, mas se tentarmos executar em nosso scratch, o jogo irá pausar antes que consigamos tocar na cor laranja, isso porque o Scratch irá fazer apenas uma leitura do código, checar se este argumento (em azul) é verdadeiro, e então encerra o jogo.

Neste exemplo temos 6 tipos diferentes de blocos.



Sensores

Condicional

Loops

Inicialização

Movimento

Som



Mas então, como resolver isso?

Uma das opções é adicionar um bloco de repetição envolta dessa condição, assim, mesmo que na primeira tentativa do Scratch ler a tela o argumento não seja verdadeiro, com essa estrutura de repetição o código não irá encerrar e poderá checar novamente, assim como está abaixo, tente você mesmo também com outros blocos de repetição.

Experimente tentar trocar a repetição por algum desses blocos e apresente ao monitor em sala:

- Enquanto;
- Operador OU
- Operador E



"Mas e o Else?"

Pode se perceber que o uso exclusivo do If pode ser pouco flexível se quisermos adicionar exceções e fazer o jogo responder quando um player não estiver realizando talvez uma atividade esperada ou adicionar um novo destino ao jogador, como um caso de exceção



Mas se quisermos tornar o jogo mais interessante e interativo, como por exemplo adicionar um outro **som** enquanto o argumento do se não for satisfeito vamos finalmente usar o else. Assim, adicionando o else para a função de "Caso o ponteiro do mouse toque no gato, toque o som de um meow, caso não, toque o som de um latido". E assim, independente de onde o mouse esteja, se não estiver tocando no gato, um latido irá tocar.

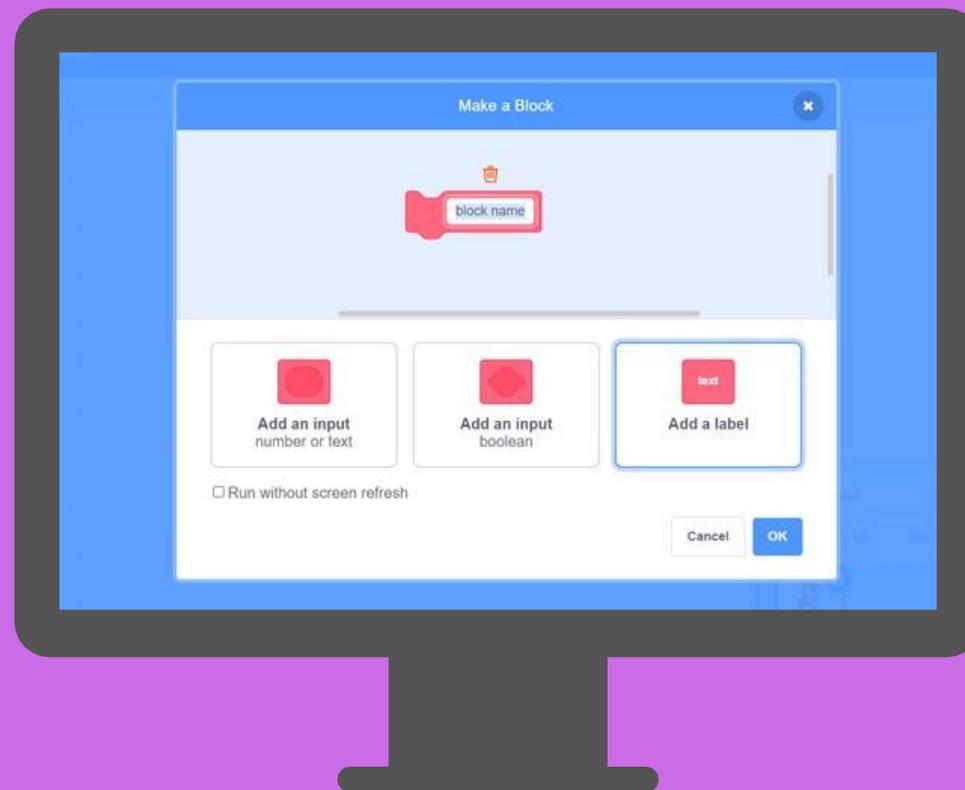


Módulo 4

Funcionalidades extras

Conteúdo

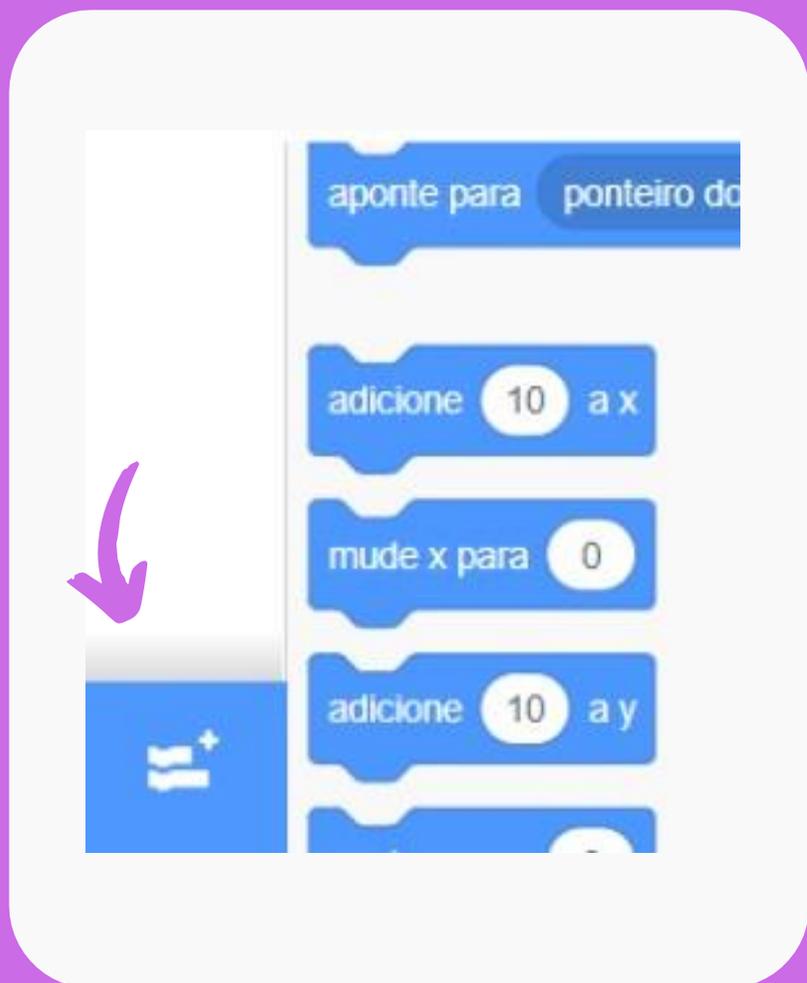
1. Extensões
2. Criando seu próprio bloco (abstrações)
3. Edição de Ator e Cenário
4. Comunidade de Projetos



Extensões

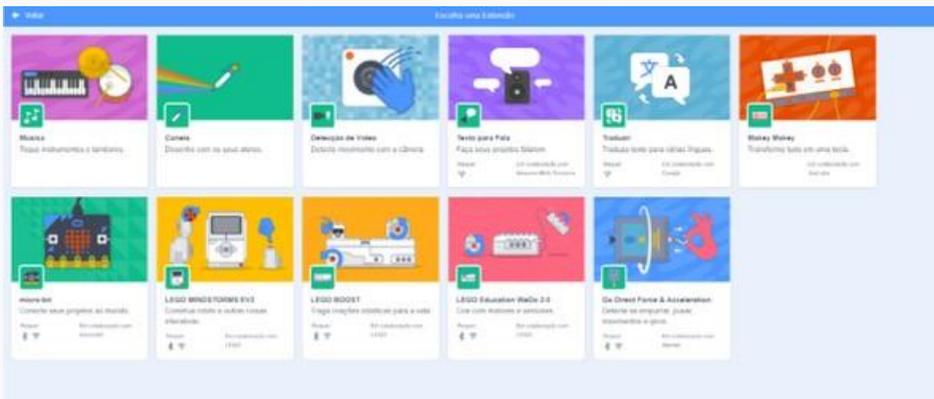


Aumentando as possibilidades do Scratch



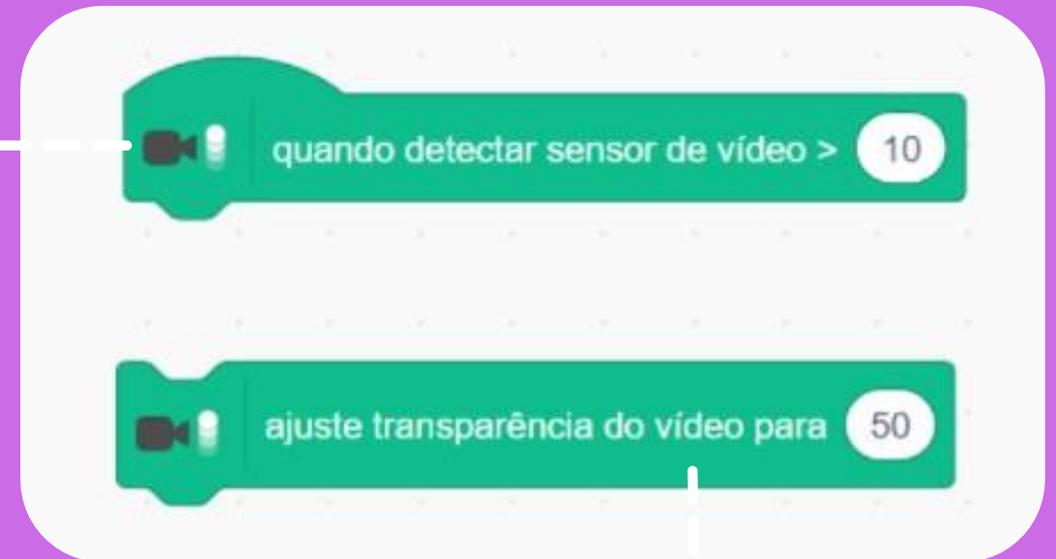
Logo no canto inferior esquerdo, existe um botão que ainda não usamos. É com ele que conseguimos acessar algumas extensões do Scratch. Mas o que são essas extensões?

São **blocos extras** que nos permitem fazer outras coisas mais interessantes, como **tocar músicas, interagir com a câmera, traduzir frases para outros idiomas e até mesmo conectar com outros dispositivos específicos**. Se você clicar no botão, verá todas as extensões disponíveis.



Clique na extensão de **Deteccão de Víde**o; vamos testar ela, mas você pode testar as outras extensões depois se quiser. Depois que clicar, novos blocos são adicionados. Vamos usar dois desses blocos:

O primeiro possui a funcionalidade de **detectar movimento perto do nosso personagem** e executar os blocos que estiverem abaixo dele. Além disso, você pode regular a **"sensibilidade"** da detecção



Já o segundo bloco muda a transparência do nosso vídeo, que estará um pouco cinza no início. Mude o valor dele para 0.

Depois que conectar os dois blocos, **qualquer bloco abaixo deles será ativado depois que detectar movimento**. Um exemplo é fazer o gato miar e mudar sua localização para algum lugar qualquer do quadro. Veja:

Caso esteja curioso, tente mudar os valores dos blocos, colocar blocos diferentes e testar novas extensões!



Criando seu próprio bloco



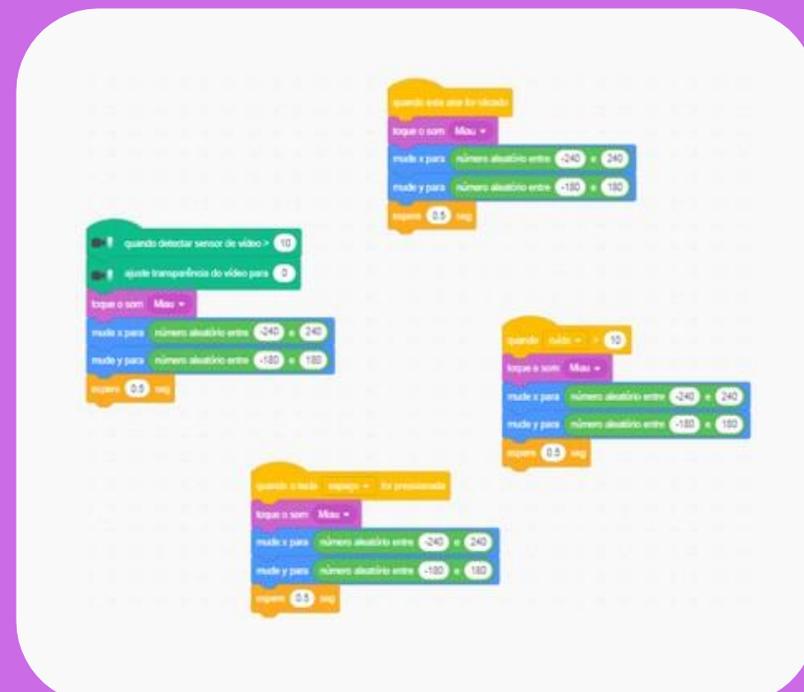
Abstraindo funcionalidades

Para pensar nessa funcionalidade, vamos usar o exemplo de fazer o gato miar e ir em algum lugar aleatório do quadro **que vimos anteriormente**.



Imagine que você quer que o gato faça tudo isso **não só quando** haja movimento na sua câmera, mas quando clicarmos com o mouse, quando a tecla espaço for pressionada, entre outros.

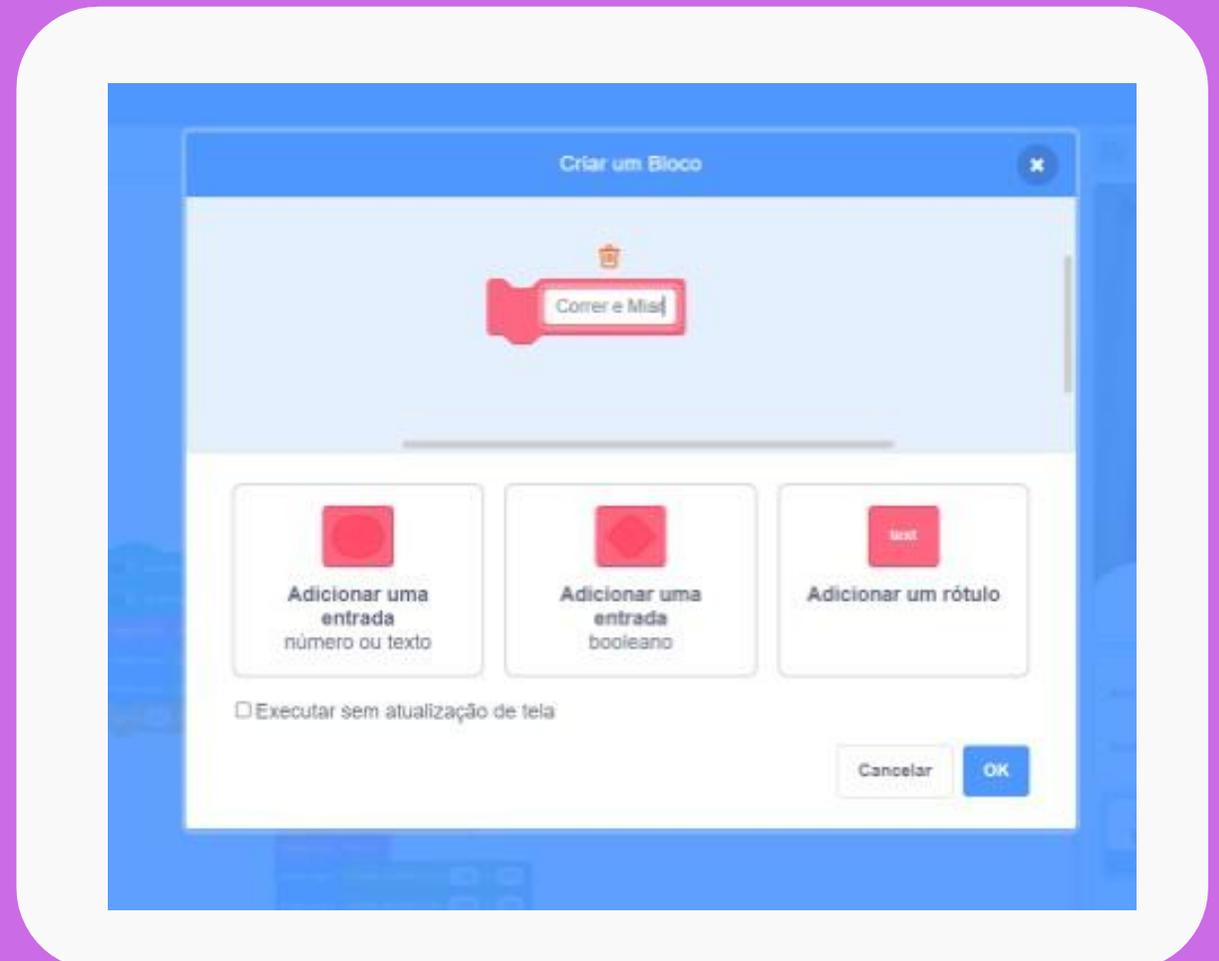
Uma forma de fazer isso seria replicar todos os blocos que fazem o gato se mover e miar várias vezes:



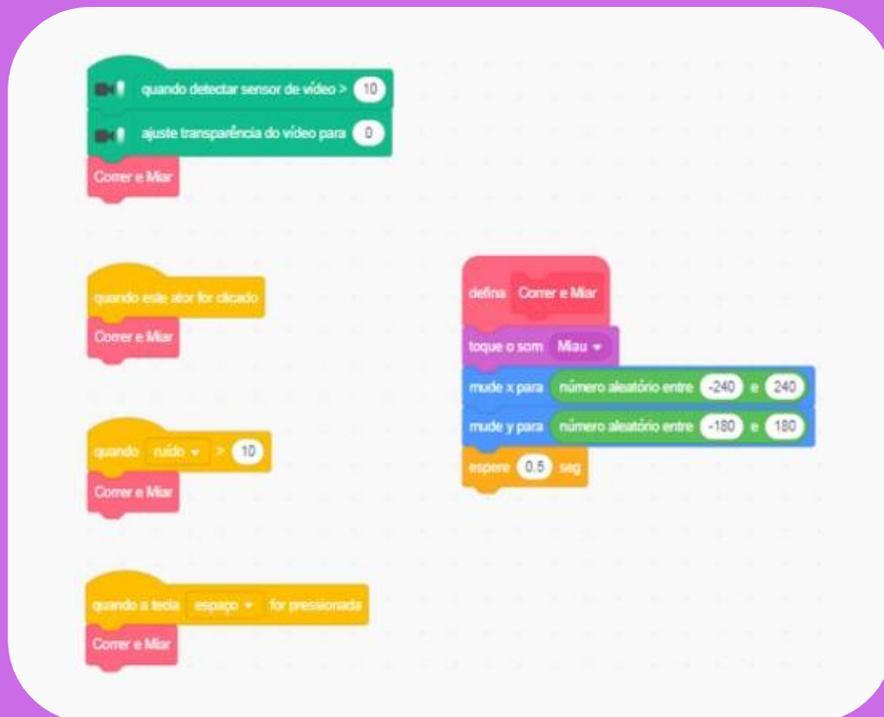
Sim, funciona!! Mas a nossa área de trabalho ficou **toda cheia e bagunçada**. Para melhorar isso, podemos criar o nosso próprio bloco responsável por fazer o gato miar e correr. Primeiro, clique em **“Meus Blocos”**, e logo em seguida, em **“Criar um Bloco”**.



Nesta nova tela, você pode escolher um nome para o bloco. Que tal “Correr e Miar”?



Depois, basta encaixar **abaixo deste novo bloco escrito "defina Correr e Miar"** o que você quer que faça parte deste novo bloco. No nosso caso, vamos fazer deste jeito:



Com o nosso novo bloco, basta **substituir os blocos repetitivos com o bloco que criamos.**

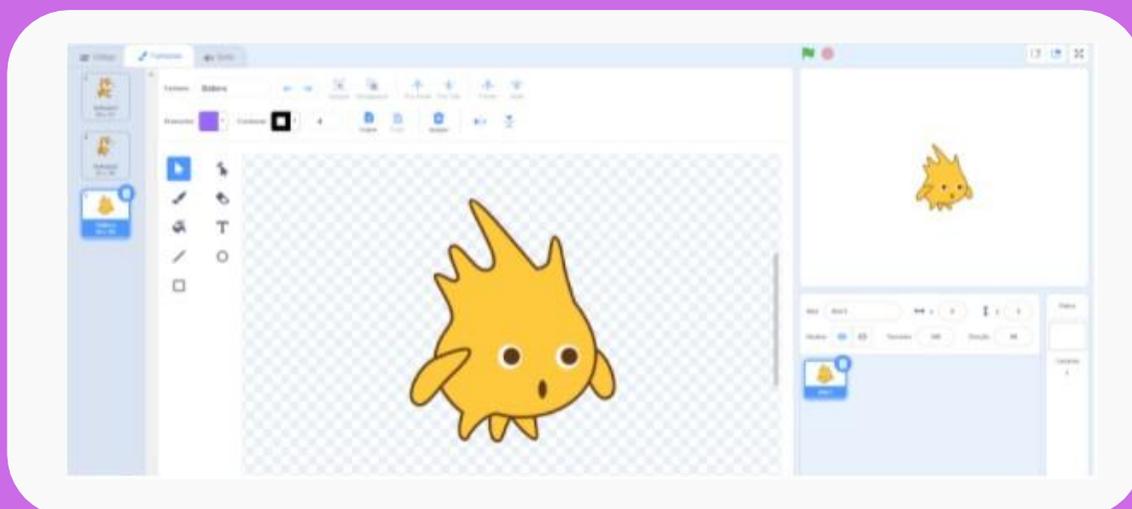
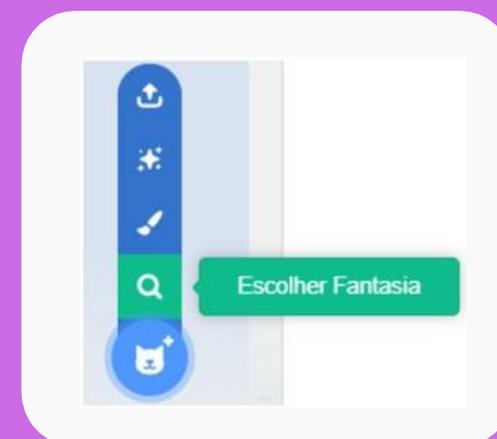
Agora, nossa área de trabalho ficou **mais limpa, organizada e fácil de entender.**

Edição de Ator e Cenário



Customizando seu Scratch

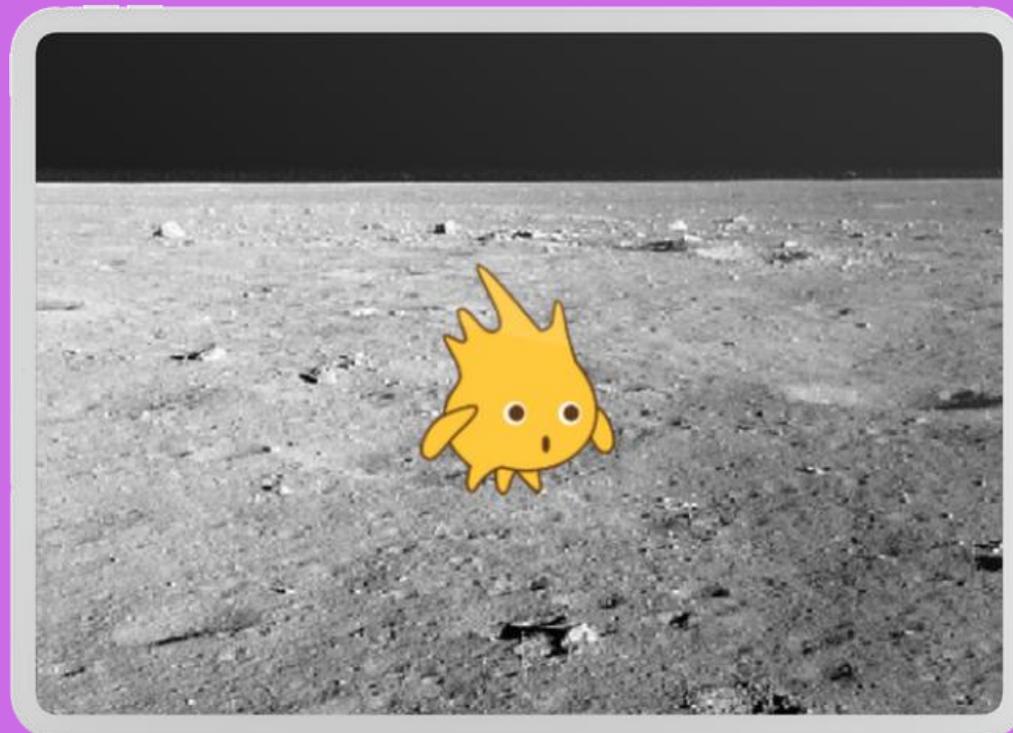
Nas abas de cima, sempre estivemos na aba “Código”, mas agora vamos ir para a seção “**Fantasia**”, onde podemos editar o nosso personagem. Dentro desta aba, no canto inferior esquerdo, temos um botão. Clique em “**Escolher Fantasia**”.



Neste painel você pode pesquisar por diversos outros personagens e **selecionar o que você mais gostar**. Aqui por exemplo, selecionamos este novo personagem, que agora também aparece no nosso quadro.

Além do seu personagem, você também pode mudar o cenário clicando no botão que fica no canto inferior direito, em **"Selecionar Cenário"**.

Novamente, **basta escolher o cenário que mais gostar.**



Comunidades de Projetos



Conhecendo outros projetos

Além de uma plataforma para poder aprender programação, o Scratch também possui uma **comunidade cheia de outros usuários que compartilham seus próprios projetos**, jogos e histórias. Para acessar estes outros projetos, devemos acessar a página principal do scratch (<https://scratch.mit.edu/>). No menu de cima, clique em **“Explorar”**.



Criar

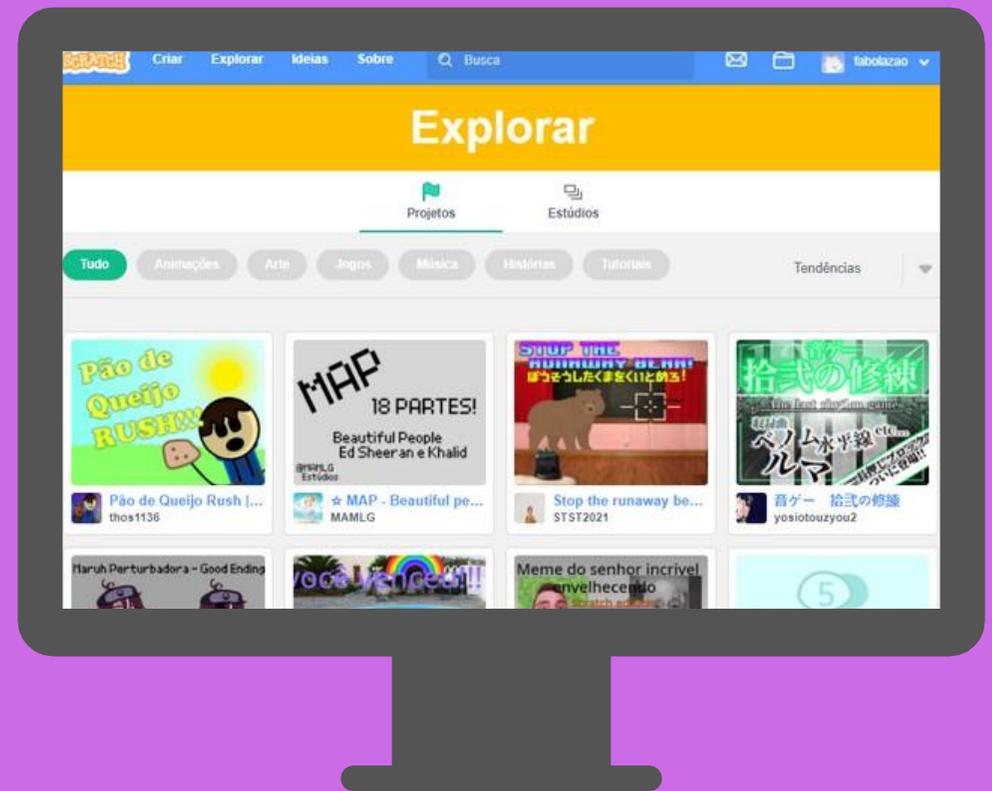
Explorar

Ideias

Sobre

Veja que existem diversos outros projetos já prontos feitos por outras pessoas que compartilharam suas ideias. **Procure algum projeto** que achar interessante e **clique nele** pra ver como funciona!

Além de ver outros projetos, você também pode salvar os projetos que fizer criando uma conta no Scratch, clicando no botão da parte de cima "Inscreva-se". Depois que criar sua conta, você poderá **salvar seus projetos e compartilhar** para outras pessoas se quiser.



Não se contente com apenas estas funcionalidades, o Scratch possui diversas outras opções e botões para explorar, além de estar sempre atualizando. Boa exploração!



Oficinas Ingrediente X

Material didático desenvolvido pela equipe do **Projeto Oficinas Ingrediente X - Programando com Scratch** do IFPE Campus Recife aprovado no edital IFES NO. 01/2021 (APOIO À INICIAÇÃO TECNOLÓGICA COM FOCO NO ENSINO DE PROGRAMAÇÃO APLICADA)

Agradecimentos e apoio:



Coordenadora - Aida Araújo Ferreira

Equipe - Welton Pereira da Luz Felix, Fábio de Lima Ferreira Papais, Ryan Gomes Paiva, Estevão Pereira da Silva, Luciano Alves Filho, Irwing Joshua Nery da Silva, Cristiane Lucia da Silva, Gilmar Gonçalves de Brito, Ioná Maria Beltrão Rameh Barbosa e Vânia Soares de Carvalho

Revisora ortográfica e gramatical - Cristiane Lucia da Silva

1ª edição
2022